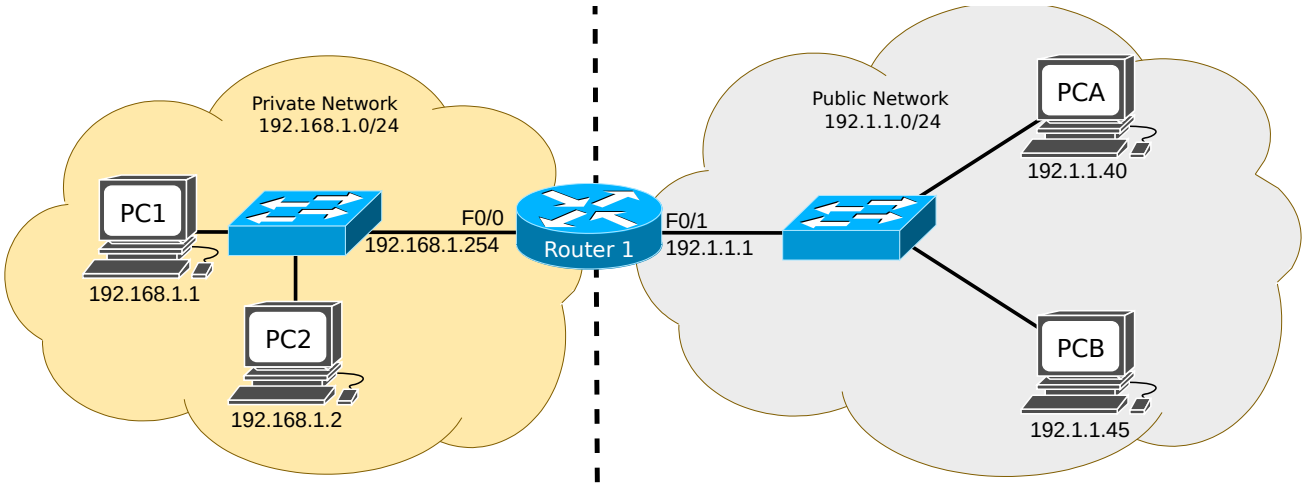
Fundamentos de Redes

Objectives

- Study of the NAT/PAT mechanisms.
- Study of DHCP.
- Study of IPv6

Dynamic NAT

1. Assemble and configure (using the GNS3 and VPCS hosts) the network depicted in the following figure which represents a small company network. The company decided to configure IP private addressing using the network 192.168.1.0/24 and NAT mechanism (without PAT) to manage all Internet accesses. IP addresses and the respective gateway addresses must be manually configured. The company only have 2 public addresses (192.1.1.1/24 and 192.1.1.21/24). **As Router 1 use the model 7200.**



Dynamic NAT Configuration

In order to define a pool of global addresses to be allocated by the dynamic NAT process, issue the following command on Router 1:

```
Router1(config)# ip nat pool MYNATPOOL 192.1.1.21 192.1.1.21 netmask 255.255.255.0
```

that defines a pool with a single public address (192.1.1.21).

The name MYNATPOOL is the name of the address pool. The first 192.1.1.21 in the command is the first IP address in the pool and the second 192.1.1.21 is the last IP address in the pool (this command creates a pool that contains only a single address).
Next, configure a standard access list to define which internal source addresses can be translated. Since any users on the private network are being translated, use the following command:

```
Router1(config)# access-list 2 permit 192.168.1.0 0.0.0.255
```

To establish the dynamic source translation, link the access list to the name of the NAT pool, as shown in the following:

```
Router1(config)# ip nat inside source list 2 pool MYNATPOOL
```

Finally, specify an interface on Router to be used by inside network hosts requiring address translation:

```
Router1(config)# interface f0/0
                #change the interface name to the one used in your router
Router1(config-if)# ip nat inside
```

Also specify an interface to be used as the outside NAT interface as follows:

```
Router1(config)#interface f0/1
                #change the interface name to the one used in your router
Router1(config-if)#ip nat outside
```

2. Start a packet capture on the public network and another on the private network. At PC1 execute a ping to 192.1.1.45 and on PC2 execute a ping to 192.1.1.45. Verify (on the router) the active NAT translations and NAT activity statistics, use the commands

```
Router1# show ip nat translations
Router1# show ip nat statistics
```

>> Which packets had the source IP addresses translated? Explain the obtained results.

3. Execute on the router the command to clear the NAT translation table:

```
Router1# clear ip nat translation *
```

and execute again at PC2 a ping to 192.1.1.40.
>> Explain the observed results.

4. Change NAT timeout to 60 seconds and clear the NAT translations table:

```
Router1(config)# ip nat translation timeout 60
Router1# clear ip nat translation *
```

At PC1 execute a ping to 192.1.1.40 and immediately after at PC2 execute repeatedly a ping to 192.1.1.40. How much time does it take to obtain connectivity between PC2 and host 192.1.1.40.
>> Explain the observed results.

Restore NAT timeout value to 86400 seconds (24 hours):

```
Router1(config)# ip nat translation timeout 86400
```

Dynamic NAT/PAT

5. The most powerful feature of NAT is address overloading, or port address translation (PAT). Overloading allows multiple inside addresses to map to a single global address. With PAT, the NAT router keeps track of the different conversations by mapping TCP and UDP port numbers.
After defining the pool of global addresses to be allocated by the dynamic NAT process and configuring the standard access list that defines which internal source addresses can be translated, configure address overloading on Router with the following command:

```
Router1(config)#ip nat inside source list 2 pool MYNATPOOL overload
```

Note: You may have to reset the active NAT translations: `clear ip nat translation *`
Repeat experience 2.
>>Which are the advantages of using NAT and PAT mechanisms?

6. From PC1 (and PC2) try to establish UDP and TCP connections (ports 80 and 22) to host 192.1.1.40:

```
PC> ping 192.1.1.40 -2 -p 80                ! UDP port 80
PC> ping 192.1.1.40 -2 -p 22                ! UDP port 22
PC> ping 192.1.1.40 -3 -p 80                ! TCP port 80
PC> ping 192.1.1.40 -3 -p 22                ! TCP port 22
```

Note: The option -p must have a space after (before the port number).
>> Verify (on the router) the active NAT translations and NAT activity statistics. Explain the obtained results.

Static NAT/PAT Translations

7. Try to ping the private network machines from PCA.

8. Suppose that now you have another public IP address available (192.1.1.201), configure the router in order to allow the PCA to access PC1.
A static translation between the inside local address of an host and one of the inside global addresses can be created using the following commands:

```
Router(config)#ip nat inside source static 192.168.1.1 192.1.1.201
```

From PCA ping PC1's static public address (192.1.1.201)

```
PCA> ping 192.1.1.201
```

>> Analyze the captured packets on the private network and explain the obtained results.
>> Discuss a scenario where static NAT/PAT is required.

DHCP

9. Configure Router 1 as DHCP server for the private network. Assume that you want to dynamically assign addresses from the range 192.168.1.100 to 192.168.1.200.

```
Router1(config)# service dhcp
Router1(config)# ip dhcp excluded-address 192.168.1.1 192.168.1.99
Router1(config)# ip dhcp excluded-address 192.168.1.201 192.168.1.254
Router1(config)# ip dhcp pool 1
Router1(dhcp-config)#network 192.168.1.0 255.255.255.0
Router1(dhcp-config)#default-router 192.168.1.254
```

Use the following commands to verify the configuration and stutus of the DHCP server:

```
show ip dhcp pool
show ip dhcp server statistics
show ip dhcp binding
```

10. Start a capture on Router 1's F0/0 interface. Configure PC1 to acquire the IPv4 address dynamically:

```
PC1> ip dhcp
```

Configure PC1 to renew the IPv4 address dynamically:

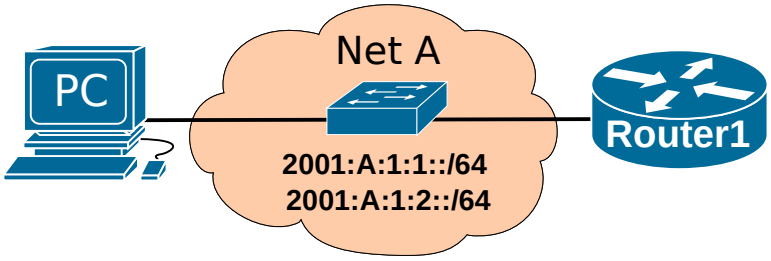
```
PC1> ip dhcp -r
```

Configure PC1 to release the IPv4 address dynamically:

```
PC1> ip dhcp -x
```

>> In each step, analyze the exchanged DHCP packets and the contents of the DHCP Bindings at Router 1.

IPv6 Basic Mechanisms



To add a VirtualBox VM Linux template to GNS3: Go to (Edit-Preferences-VirtualBox-VirtualBox templates" and create a new VM template based on an existing VirtualBox machine. Use an appliance available to download at the course page in elearning.ua.pt (login/password: labcom/labcom) or choose another appliance from http://www.osboxes.org/virtualbox-images/ .
Note1: The VM should be powered off and the network adapter should be "not attached".
Note2: Always start/stop the VM use the GNS3 interface (NOT the VirtualBox App).
Note3: To use multiple VM instances, you may clone the original machine.

11. Considering the above depicted network, start by connecting the PC (a VirtualBox VM Linux) to the switch without any other connections.
Start a capture in the link between the PC and the Switch. Find your Ethernet interface name, disable and enable the Ethernet interface and restart the network manger with:

```
ip link
sudo ip link set down enp0s3
sudo ip link set up enp0s3
sudo systemctl restart NetworkManager
```

and verify the PC's addressing information:

```
ip addr
```

Stop the capture and analyze the IPv6 packets.

12. Connect Router1 to the switch and start a capture in the link between the PC and the Switch.
Power on Router1 and configure it's interface to network A.

```
Router1(config)# ipv6 unicast-routing
Router1(config)# interface <if-name>
Router1(config-if)# ipv6 enable
Router1(config-if)# no shutdown
```

Verify router's interfaces names and configuration:

```
Router1# show ipv6 interface
Router1# show ipv6 interface brief
```

Find your Ethernet interface name and current addresses, disable and enable the Ethernet interface, and restart the network manger and verify it's new address:

```
ip addr
sudo ip link set down enp0s3
sudo ip link set up enp0s3
sudo systemctl restart NetworkManager
ip addr
```

Stop the capture and analyze the IPv6 packets and equipment's information. Use the commands:

```
show ipv6 interface brief
show ipv6 route
```

to verify interfaces' IPv6 addressing and verify router's IPv6 routing table.

13. Re-start a capture in the link between the PC and the Switch. Configure Router's interface with a manually defined IPv6 global address from network 2001:A:1:1::/64.

```
Router1(config)# interface <if-name>
Router1(config-if)# ipv6 address 2001:A:1:1::100/64
Router1(config-if)# no shutdown
```

Verify PC's Ethernet interface information. Stop the capture and analyze the IPv6 packets. Verify Router's interfaces IPv6 addresses and router's IPv6 routing table.

>> Explain the process by which the PC obtained the IPv6 addresses.

14 Re-start a capture in the link between the PC and the Switch. Configure Router's interface with a EUI-64 based IPv6 global address from network 2001:A:1:2::/64.

```
Router1(config)# interface <if-name>
Router1(config-if)# ipv6 address 2001:A:1:2::/64 eui-64
Router1(config-if)# no shutdown
```

Verify PC's Ethernet interface information. Stop the capture and analyze the IPv6 packets. Verify Router's interfaces IPv6 addressing and the router's IPv6 routing table

>> Explain the process by which the Router completed the last 64 bits of its IPv6 addresses.

>> Discuss a possible disadvantage of using the standard EUI-64 at routers' interfaces.

>> Does the process by which the PC obtained the IPv6 addresses, changed by using the EUI-64 standard at the Router.

15. Re-start a capture in the link between the PC and the Switch. At the PC, using the command *ping6* perform a ping to:

a) Router's Link-Local address (you need to define the output interface with option "`-I eth0`" or "`-I enp0s3`").

b) Router's Global address from network 2001:A:1:1::/64.

c) Router's Global address from network 2001:A:1:2::/64.

Stop the capture and analyze the IPv6/ICMPv6 packets.

>> Explain the physical addresses resolution process in IPv6.